

논문 2005-42SD-6-6

# 통신 시스템을 위한 고성능 재구성 가능 코프로세서의 설계

## (Novel Reconfigurable Coprocessor for Communication Systems)

정 철 윤\*, 선우 명 훈\*\*

(Chul Yoon Jung and Myung Hoon Sunwoo)

### 요 약

본 논문은 통신 시스템에서 요구하는 다양한 연산과 고속의 동작을 수행할 수 있는 재구성 가능 코프로세서를 제안 하였다. 제안된 재구성 가능 코프로세서는 스크램블링, 인터리빙, 길쌈부호화, 비터비 디코딩, FFT 등과 같은 통신 시스템에 필수적인 연산 동작을 쉽게 구현할 수 있는 특징을 가진다. 제안된 재구성 가능 코프로세서는 VHDL로 설계하여 SEC 0.18 $\mu$ m 표준셀 라이브러리를 이용해 합성하였으며, 총 35,000 게이트에 3.84ns의 최대 동작 속도를 보였다. 제안된 코프로세서에 대한 성능검증 결과 IEEE 802.11a WLAN 표준에 대해 기존 DSP에 비해서 FFT 연산과 Complex MAC의 경우 약 33%, 비터비 디코딩의 경우 약 37%, 스크램블링 및 길쌈부호화의 경우 약 48%~84%의 연산 사이클 감소를 확인하였으며 다양한 통신 알고리즘에 대해 기존 DSP보다 우수한 성능을 나타내었다.

### Abstract

This paper proposes a reconfigurable coprocessor for communication systems, which can perform high speed computations and various functions. The proposed reconfigurable coprocessor can easily implement communication operations, such as scrambling, interleaving, convolutional encoding, Viterbi decoding, FFT, etc. The proposed architecture has been modeled by VHDL and synthesized using the SEC 0.18 $\mu$ m standard cell library. The gate count is about 35,000 gates and the critical path is 3.84ns. The proposed coprocessor can reduced about 33% for FFT operations and complex MAC, 37% for Viterbi operations, and 48% ~ 84% for scrambling and convolutional encoding for the IEEE 802.11a WLAN standard compared with existing DSPs. The proposed coprocessor shows performance improvements compared with existing DSP chips for communication algorithms.

**Keywords :** Reconfigurable Processor, VLSI architecture, Software Defined Radio (SDR), Digital Signal Processor

## I. 서 론

최근 통신서비스의 성장은 많은 경쟁적이고 비호환적인 표준들을 등장 시켰다. 이러한 표준에는 xDSL (Digital Subscriber Line)<sup>[1, 2]</sup>, WLAN (Wireless Local Area Network)<sup>[3]</sup>, DAB (Digital Audio Broadcast)<sup>[4]</sup>, DVB (Digital Video Broad-cast)<sup>[5]</sup>, IMT-2000

(International Mobile Telecommunications-2000) 등이 있다. 이들 통신 표준들은 유사한 연산을 가지고 있음에도 서로 다른 연산 방식을 필요로 함에 따라 표준마다 별도의 알고리즘 및 하드웨어가 필요하였다. 이러한 문제점을 해결하기 위해 다양한 표준과 기존 시스템과의 호환성을 고려한 SDR(Software Defined Radio)과 같은 유연한 하드웨어 구조에 대한 연구가 활발히 이루어지고 있다<sup>[6]</sup>. 멀티모드, 멀티밴드를 지원하기 위한 SDR 개념의 하드웨어 연구를 기반으로 다양한 통신 표준을 모두 수용할 수 있는 차세대 통신 시스템을 위한 유연한 구조의 하드웨어 개발에 대한 요구가 증대되고 있으며 이러한 요구를 충족시키기 위한 프로그램 가능 소자의 개발이 특히 주목받고 있다.

그러나 최근의 FPGA 혹은 DSP 기반의 프로그램 가

\* 정회원, \*\* 평생회원 아주대학교 전자공학부  
(School of Electrical and Computer Eng., Ajou Univ.)

※ 본 연구는 산업자원부와 과학기술부에서 시행하는 시스템 IC 2010 사업과 과학기술부에서 시행하는 국가 지정 연구실 (NRL) 사업, ITRC의 HY-SDR 연구센터 및 IDEC 사업의 지원을 받아 수행되었습니다.

접수일자: 2005년2월14일, 수정완료일: 2005년5월27일

능 및 재구성 가능 구조에 대한 개발은 효율성과 성능면에서 다양한 표준안의 요구를 만족시키지 못하고 있다. 따라서 DSP와 FPGA 정도의 프로그램 능력을 가지면서 고성능, 저전력 특성을 갖는 소자의 개발이 필요하다. 앞으로의 통신용 프로세서는 이러한 문제들을 해결할 수 있는 RP (Reconfigurable Processor), ASP (Application Specific Processor), ASIP (Application Specific Instruction Processor) 등과 같은 적절한 유연성과 고성능을 가지는 소자들을 결합한 고성능의 재구성 가능 플랫폼 형태가 될 것으로 예상된다<sup>[7]</sup>. 이에 본 논문은 다양한 통신 표준을 수용할 수 있는 차세대 통신 시스템에 적합한 재구성 가능 프로세서의 구조를 제안하고 제안된 구조의 설계 및 성능평가를 수행하였다.

본 논문은 다음과 같이 구성되어 있다. II장에는 재구성 가능 코프로세서의 연산기 구조를 제안하고, III장에서는 재구성 가능 코프로세서를 제안한다. IV장에서는 재구성 가능 코프로세서의 구현 결과 및 기존 DSP들과의 성능 비교를 기술하였다. 마지막으로 V장에서 결론을 기술하였다.

## II. 재구성 가능 코프로세서의 연산기 구조

차세대 통신 시스템을 위한 하드웨어 구조는 적절한 유연성과 고성능, 저전력 특성을 요구하고 있다. 하지만 이러한 특성들의 경우 하드웨어 구현 시 각각이 상호 상반되게 작용하여 하드웨어 개발의 제약요인이 되고 있다. 따라서 기존 설계 방식과 하드웨어 구조를 통해서는 차세대 통신 시스템에서 요구하는 특성을 모두 만족시키기 어려운 것이 사실이다. 이에 본 연구에서는 다양한 통신표준에 대한 분석을 통해 SDR에서 이야기하고 있는 재구성 가능한 프로세서를 기반으로 다양한 통신표

준을 만족할 수 있는 고성능의 재구성 가능 통신 프로세서구조 제안 및 성능 분석을 수행하였다.

### 1. 통신 시스템에 필요한 기본 연산 분석

일반적인 통신 시스템에서 데이터는 소스 코딩, 채널 코딩, 변조, 필터링 등의 과정을 거쳐 채널로 전송되고, 수신 쪽에서는 필터링, 동기, 복조, 채널 디코딩 등의 과정을 거쳐 데이터를 복원한다. 최근 각광받고 있는 통신 방식인 CDMA (Code Division Multiplexing Access)와 OFDM (Orthogonal Frequency Division Multiplex) 통신 시스템에서는 각각 확산과 다중 반송파 변조의 과정을 더 거친다. 그림 1은 이러한 과정을 간략히 나타내고 있다.

각 과정에 대하여 다양한 통신 표준에 따라 사용되는 방식을 살펴보면 다음과 같다. 채널 코딩의 방법으로는 스크램블링(Scrambling), 길쌈 부호화(Convolutional Encoding), 리드솔로몬(Reed-Solomon) 인코딩, 터보 인코딩(Turbo Encoding), 인터리빙(Interleaving) 등의 방법이 널리 사용된다. 수신쪽은 송신쪽과 대응되는 디스크램블링, 비터비 디코딩 (Viterbi Decoding), 리드솔로몬 디코딩, 터보 디코딩, 디인터리빙 등이 사용된다. 변조 방식으로는 QPSK(Quadrature Phase Shift Keying), QAM(Quadrature Amplitude Modulation) 등의 방식이 주로 사용된다. 다중 반송파 변조를 위해서는 IFFT (Inverse Fast Fourier Transform)와 FFT가 사용되고, 확산 및 역확산을 위해서는 의사 잡음 코드 (Pseudo Noise Code)를 데이터에 곱하는 확산 및 역확산이 사용된다. 이러한 각 기능 블록들은 표준에 따라 그 파라미터가 다르지만 기본적인 연산은 동일하다. 따라서 이러한 기본적인 공통 연산을 특징에 따라 분류하여 다음과 같이 정리할 수 있다<sup>[8]</sup>.

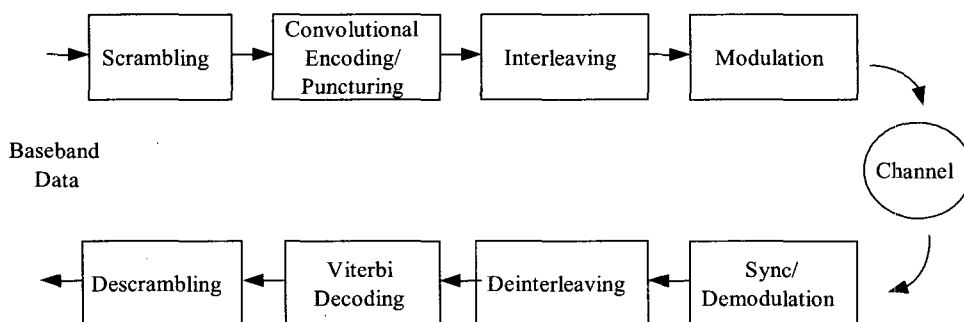


그림 1. 통신 시스템 구성도  
Fig. 1. Illustration of communication systems.

가. 곱셈 및 덧셈/뺄셈 연산

FFT, FIR (Finite Impulse Response) 필터, 적응 필터 (Adaptive Filter), 컨볼루션 (Convolution), 상관 (Correlation), 동기 (Synchronization), DCT (Discrete Cosine Transform), 벡터 연산, 비터비 디코딩, 터보 디코딩 등<sup>[9]</sup>.

나. 쉬프트 및 논리 연산

리드솔로몬 인코딩, 리드솔로몬 디코딩, 터보 인코딩, CRC(Cyclic Redundancy Check) 생성, 스크램블링, 의사잡음 코드 생성, 길쌈 부호화 등.

다. 비트조작 연산

인터리빙, 펄칭(Puncturing), Bit Unpack, 변조, 비트열 다중화(Bit Stream Multiplexing) 등.

이와 같이 통신 시스템에 널리 사용되는 연산들은 3가지의 기본적인 공통 연산들로 분류할 수 있으며 이러한 기본연산을 효율적으로 지원하는 연산기들을 사용함으로써 유연성과 고성능이라는 두 가지 요구를 모두 만족시킬 수 있다.

2. 통신 시스템을 위한 재구성 가능 하드웨어 구조

앞에서 분류한 공통의 연산들을 수용할 수 있는 하드웨어 구조 설계를 통하여 다양한 통신 시스템에서 요구되는 연산을 수행할 수 있는 재구성 가능한 하드웨어 구

조를 제안하였다. 이러한 재구성 가능한 각 하드웨어 구조에 대해 살펴보면 다음과 같다<sup>[10, 11]</sup>.

2.1 곱셈 및 덧셈/뺄셈 연산 구조 제안

곱셈 및 덧셈 연산은 FFT, 필터링, 동기 및 상관 연산에 주로 사용된다. 또한 덧셈, 뺄셈 및 비교 연산은 비터비 디코딩, 터보 디코딩에 주로 사용된다. 곱셈과 덧셈 연산은 거의 같이 사용되기 때문에 하나의 기능블록으로 구현하는 것이 효율적이다. 그림 2는 곱셈 및 덧셈 연산으로 가능한 연산들을 나타내고 있다. 그 종류에는 실수 곱셈(Real Multiply), 실수 곱셈누적(Multiply and Accumulate), 실수와 복소수의 곱셈(Real X Complex Multiply and Accumulate), 복소수 곱셈(Complex Multiply), 복소수 곱셈누적(Complex Multiply and Accumulate), FFT 버터플라이(Butterfly)가 있다.

위의 연산 중에 FFT 버터플라이가 가장 많은 곱셈과 덧셈을 필요로 한다. 따라서 FFT 버터플라이의 부분 연산을 지원할 수 있는 연산구조를 가지고 앞서 말한 다양한 곱셈 및 덧셈을 수행할 수 있는 구조를 설계할 경우보다 효율적인 연산기를 구성할 수 있다.

제안된 덧셈/뺄셈기는 FFT, MAC 등의 연산을 위한 하나의 32비트 연산을 지원할 수 있고 BM(Branch Metric), PM(Path Metric) 연산을 위해 두개의 16비트 연산을 지원할 수 있다. 그림 3은 이러한 연산을 지원하기 위한 연산기 구조로써 4개의 16비트 곱셈기와 6개의 32비트 덧셈/뺄셈기로 구성된다. 제안된 구조는 화살표

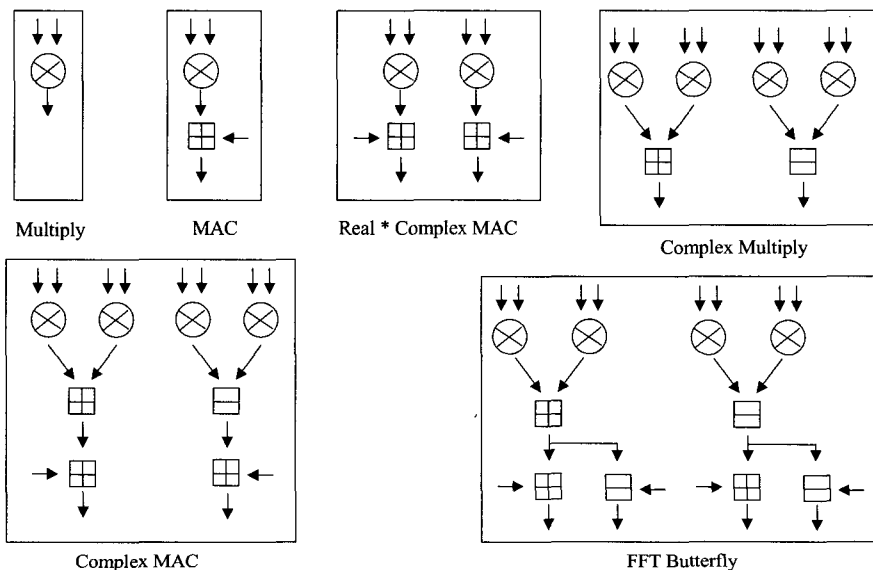


그림 2. 곱셈 및 덧셈으로 가능한 연산들  
Fig. 2. General operations based on multiply and addition.

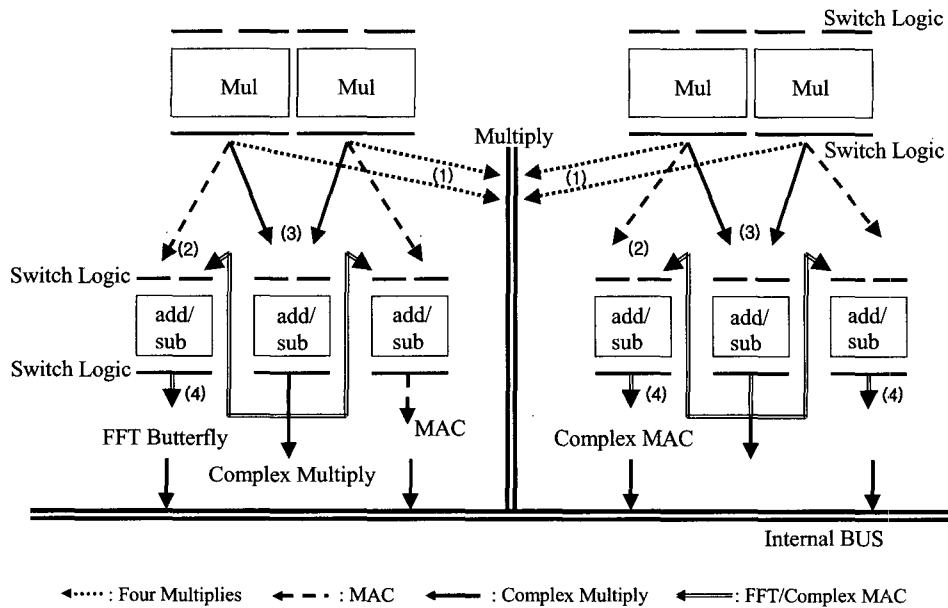


그림 3. 제안한 곱셈 및 덧셈 구조  
 Fig. 3. Proposed multiplier/adder architecture.

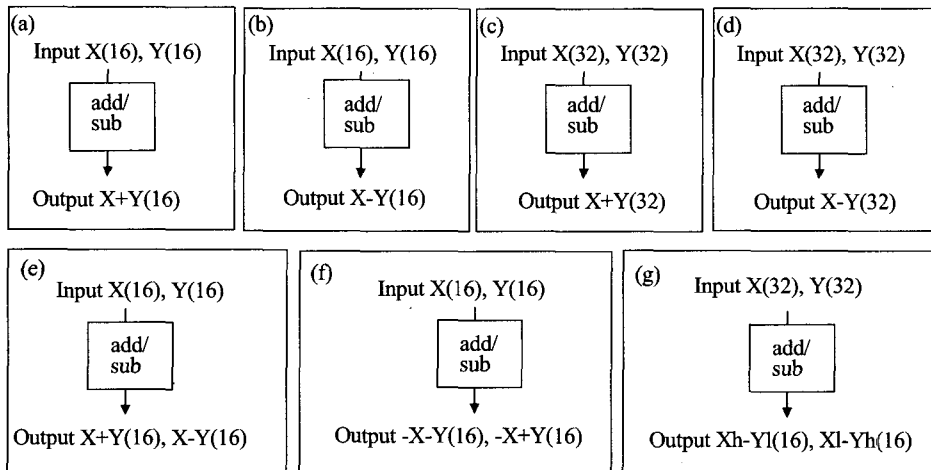


그림 4. 제안한 덧셈 및 뺄셈 구조  
 Fig. 4. Additions and subtractions using the proposed structure.

(1)을 따라서 4개의 16비트 곱셈연산을 지원할 수 있고 화살표 (2)를 따라서 4개의 MAC연산을 수행할 수 있다. 화살표 (3)을 따라 4개의 복소 곱셈 연산과, 화살표 (4)를 따라서 하나의 FFT 연산 및 두개의 복소 MAC 연산을 수행할 수 있다.

그림 4는 제안된 구조를 이용한 다양한 덧셈 및 뺄셈 연산을 표현한 것이다. 그림 4(a)와 그림 4(b)는 하나의 16비트 덧셈/뺄셈 연산을 간략하게 나타낸 것이다. 그림 4(c)와 그림 4(d)는 하나의 32비트 덧셈/뺄셈 연산을 나타낸 것이다. 그림 4(e)와 그림 4(f)는 두개의 16비트 덧셈/뺄셈 연산을 표현하였다. 그림 4(g)는 2개의 32비트 입력을 받아 한쪽 입력의 상위 16비트와 다른쪽 입력의

하위 16비트를 빼어 16비트 출력을 만들고, 한쪽 입력의 하위 16비트와 다른쪽 입력의 상위 16비트를 빼어 16비트 출력을 만드는 것을 표현하였다.

그림 4의 덧셈/뺄셈은 비터비 디코딩, 터보 인코딩 등과 같은 다양한 동작을 지원할 수 있다. 비터비 디코딩의 경우 일반적으로 BM 연산, ACS(Add Compare Select) 연산, 생존자 경로 연산의 3가지 과정으로 구성되어 있다. BM 연산은 트렐리스 구조를 통하여 수신 데이터와 코드워드 사이의 거리를 계산하여 BM을 생성한다.

그림 5는 코드율 1/2과 1/3인 경우의 일반적인 비터비 BM 연산 방정식을 나타낸다. 이러한 BM 연산과정을

$\begin{aligned} Bm00 &= (-x) + (-y) \\ Bm01 &= (-x) + y \\ Bm10 &= x + (-y) \\ Bm11 &= x + y \end{aligned}$ <p>(a) Code Rate = 1/2</p>	$\begin{aligned} Bm000 &= (-x) + (-y) + (-z), & Bm100 &= x + (-y) + (-z) \\ Bm001 &= (-x) + (-y) + z, & Bm101 &= x + (-y) + z \\ Bm010 &= (-x) + y + (-z), & Bm110 &= x + y + (-z) \\ Bm011 &= (-x) + y + z, & Bm111 &= x + y + z \end{aligned}$ <p>(b) Code Rate = 1/3</p>
-----------------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

그림 5. 가지 메트릭 계산식  
Fig. 5. General Viterbi BM equations.

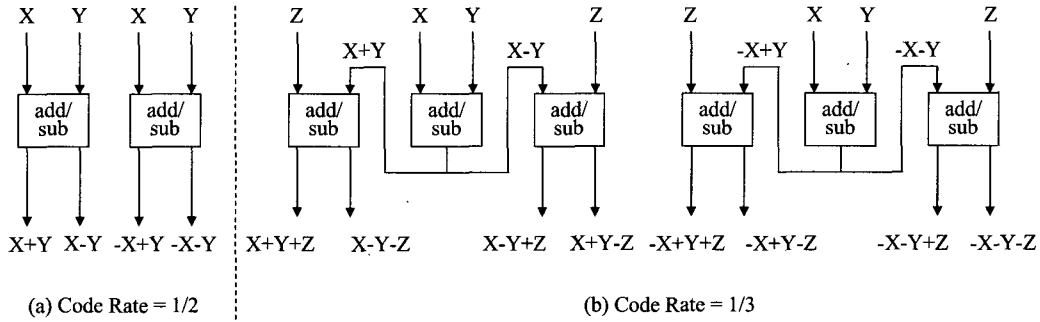


그림 6. 제안된 구조를 이용한 가지 메트릭 연산  
Fig. 6. BM calculation using the proposed structure.

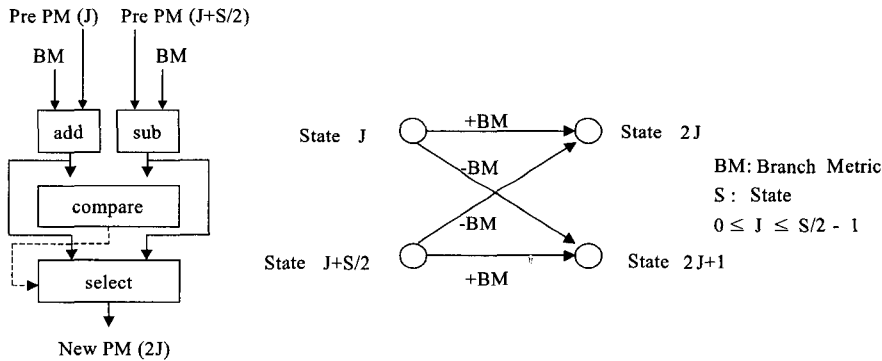


그림 7. 경로 메트릭 계산과 트렐리스 버터플라이  
Fig. 7. PM and trellis butterfly function.

살펴보면 일련의 덧셈/뺄셈 연산으로 구성되어 있음을 확인할 수 있다. 따라서 이러한 연산을 제안된 하드웨어 구조를 이용해 구현하였다. 그림 6은 제안된 하드웨어 구조를 이용한 BM 연산을 나타낸 것으로 코드율 1/2일 경우 그림 4(e)와 4(f)를 사용하여 구현하고, 코드율 1/3 일 경우 그림 4(e)와 4(f)를 사용하여 6개의 덧셈/뺄셈 연산을 통해 구현할 수 있다.

그림 7은 일반적인 ACS 연산과 트렐리스 버터플라이를 나타낸 것이다. ACS 연산의 경우 그림 7에 나타낸 것과 같이 BM 값과 PM 값과의 덧셈 연산으로 구성되어 있다. 그림 8은 제안된 하드웨어 구조를 이용한 트렐리스 버터플라이 연산을 나타낸 것으로 PM 값에 BM 값을 더하기 위해 그림 4(e)를 수행하는 동작과 연산되어진 새로운 두 PM 값을 비교하는 그림 4(g)를 수행하

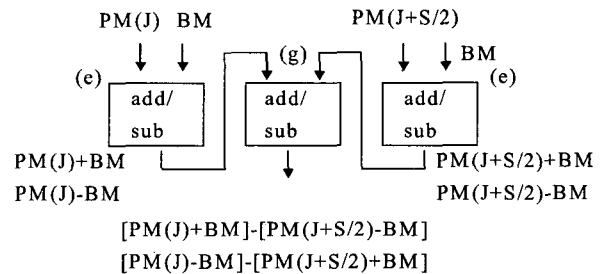


그림 8. 제안된 구조를 이용한 트렐리스 버터플라이 연산  
Fig. 8. Trellis butterfly calculation using the proposed structure.

는 동작으로 구성되어 있다. 새로운 PM 값은 그림 7의 뺄셈 결과의 부호에 따라 2J와 2J+1 상태 메모리에 저장된다.

제안된 연산기 구조는 SEC 0.18 $\mu$ m 표준셀 라이브러리를 이용하여 250MHz 동작 속도에 2,500 게이트를 갖는 16x16 곱셈기와 540MHz 동작속도에 1,100 게이트를 갖는 32비트 덧셈기로 설계되었다. 따라서 제안한 하드웨어 구조는 다양한 통신 알고리즘을 수용할 수 있으면서 고속의 동작을 지원할 수 있을 것으로 예상된다.

## 2.2 비트조작 연산기 구조

본 절에서는 스크램블링, 길쌈부호화, 평처링, 인터리빙 등의 비트조작 연산기 구조에 대하여 기술하였다. 일반적으로 스크램블링, 평처링과 같은 비트조작 연산은 일정한 패턴을 가지고 있지만 저마다 다른 비트조작 연산 특성으로 인하여 다양한 통신 표준의 요구를 모두 만족시킬 수 있는 하드웨어를 구현하기는 쉽지 않다. 본 연구에서는 이러한 비트조작 연산의 가장 기본이 되는 특정 위치의 비트 추출 혹은 비트 삽입 등의 연산을 다양한 연산 요구에 따라 유연하게 사용 가능한 하드웨어 구조에 대한 연구를 진행하였으며, 참고문헌 [12]에서 제안한 다양한 통신 표준에서 요구하는 비트조작 연산을 지원할 수 있는 재구성 가능한 비트조작 연산기 구조를 코프로세서에 적용하였다. 제안된 비트조작 연산기 구조는 1~2 사이클 내에 비트 삽입/추출 연산과 길쌈부호화 및 스크램블링 연산을 완료할 수 있도록 하였으며 다양한 표준의 요구에 맞게 원하는 비트 삽입/추출 위치를 가변적으로 조절 가능한 기능을 제공하고 있다.

그림 9는 시프트/XOR 어레이와 비트 삽입/추출기, 비트단위 입력가능 레지스터를 포함하고 있는 제안된 비트조작 연산기 구조를 표현하였다. Mask1과 Mask2 신호는 비트조작 연산기를 제어하기 위해 사용되는 신호로 두 값을 이용하여 비트 삽입/추출 비트의 위치를 결정하고 동작을 수행할 수 있도록 하였다. 비트조작 연산기 구조 및 동작에 대한 자세한 설명은 참고문헌 [12]에 자세히 기술되어 있다.

## III. 재구성 가능 코프로세서

앞 절에서는 통신 시스템에서 보편적으로 사용되는 알고리즘에 적합한 재구성 가능 하드웨어 구조들을 제안하였다. 본 절에서는 II절에서 제안한 하드웨어 구조들을 적용한 재구성 가능 코프로세서를 제안한다. 그림 10은 제안된 재구성 가능 코프로세서를 타나낸 것으로 연산기와 메모리 주소 생성(Address Generation Unit) 및 데이터 버퍼링(Buffer)을 하는 메모리 인터페이스, 메모

리에서 받은 명령어를 해석(Decoding)하거나 재구성 정보를 생성하는 재구성 매니저(Configuration Manager, CM), 각 연산기들에 대한 제어와 구성정보를 저장하고 있는 구성 레지스터(Configuration Register, CR)로 구성된다.

재구성 가능 코프로세서는 다음과 같은 동작 과정으로 이루어진다. 마이크로프로세서는 재구성 가능 코프로세서의 동작 구조를 결정하기 위해 인코딩된 명령어를 CM에 제공하게 된다. 제공된 명령에 따라 각 연산 유닛에 대한 재구성 정보를 재구성 메모리로부터 읽어오게 된다. 재구성 정보는 각 연산 유닛의 경로 구조 및 메모리 접속 대역폭, 연산 종류 및 연산 유닛간의 상호 연결 정보를 가지고 있다. CM은 이러한 정보를 CR에 저장시키고, 제공된 데이터와 명령어에 따른 연산을 수행하게 된다. 이러한 재구성 가능 프로세서 구조를 사용할 경우 기존 DSP의 고정된 데이터 경로 구조로 인한 불필요한 연산 사이클 증가 및 연산 유닛을 다 사용하지 못하는 문제를 해결할 수 있을 뿐만 아니라 통신표준 변화에 따른 연산방식 변화에 대해 재구성 데이터의 수정만으로 빠른 적응이 가능하게 된다.

그림 11은 재구성 가능 코프로세서의 CM을 도시한다. CM은 코프로세서의 동작과 재구성을 통제하는 기능을 수행하는 것으로, 재구성 정보 해석기(Configuration Information Decoder)와 재구성 신호 생성기(Configuration signal generator), CR로 구성되어 있다. 재구성 정보 해석기는 프로세서로부터 받은 인코딩된 명령어를 해석하고 재구성 메모리의 주소를 생성한다. 재구성 신호 생성기는 각 연산 유닛간의 연결 정보와 같은 제어 정보를 생성한다. 생성된 제어 정보는 1~2 비트로 구성되어 CR에 저장됨으로써 재구성에 따른 동작을 완료하게 된다. 인코딩된 명령어의 해석 및 재구성 메모리에 접근하는데 1 사이클이 소요되고, 재구성 정보의 생성을 위해 1 사이클이 소요됨으로써 전체적으로 제안된 코프로세서의 재구성을 위해서 2 사이클을 요구하게 된다. 이는 전체 통신 연산과 비교했을 때 매우 작은 시간이라고 할 수 있다.

다양한 베이스밴드 표준안에서 메모리 구조는 중요한 요소 중에 하나이다. 본 연구에서는 재구성 가능 코프로세서의 동작을 위해 요구되는 데이터와 메모리 구조에 대한 다양한 연구를 진행하였다. 특히 FFT 알고리즘의 경우 다른 통신 알고리즘과 비교했을 때 더 복잡한 연산을 요구하게 된다. 이는 두개의 입력 데이터와 1개의 보조 연산 성분, 두개의 출력을 필요로 하게 되고, 각 데이

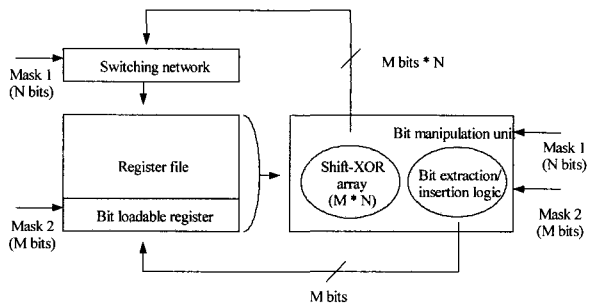


그림 9. 제안한 비트 조작 연산기 구조  
Fig. 9. Proposed bit manipulation structure.

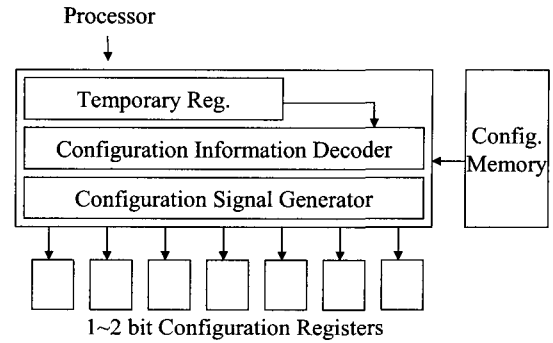


그림 11. 재구성 매니저 구조  
Fig. 11. CM structure.

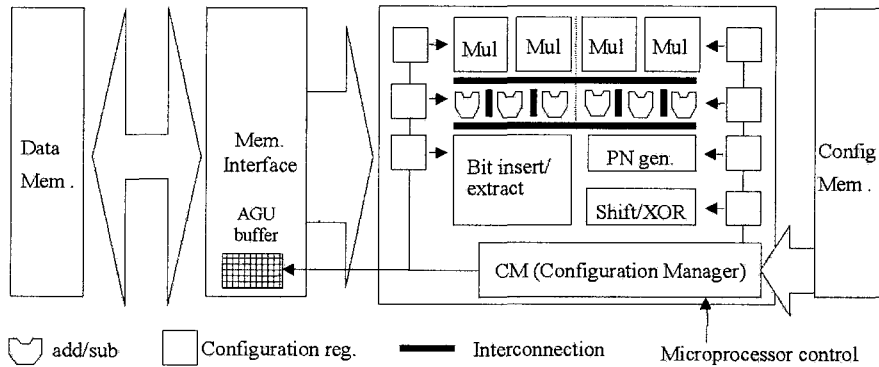


그림 10. 제안한 재구성 가능 코프로세서  
Fig. 10. Proposed reconfigurable coprocessor.

터가 다시 실수와 허수로 구성됨으로써 전체적으로 10개의 16비트 데이터를 필요로 하게 된다. 이러한 연산을 지원하기 위해 본 연구에서는 특별한 메모리 인터페이스를 설계하였다. 설계된 메모리 인터페이스는 입출력을 위해서 3개의 버스 인터페이스와 결과를 저장하기 위해 두개의 DMA 인터페이스로 구성되어 있다. 제안된 메모리 구조는 AMBA 버스 구조와 같은 멀티 레이어 버스 구조를 이용하여 쉽게 구현 가능하도록 하였다.

제안된 코프로세서는 통신 알고리즘을 분석하여 설계한 재구성 가능 하드웨어 구조를 사용하였다. 제안된 하드웨어 구조는 일반적인 DSP에 비해 효율성과 유연성에서 훨씬 우수한 성능을 나타내었다. 또한 제안된 코프로세서는 CM을 사용함으로써 메모리의 패치, 디코딩 시간을 대폭 줄일 수 있는 구조적 특징을 가지도록 하였다. 예를 들어 FFT, 비터비 디코딩 및 필터링 알고리즘은 규칙적인 연산흐름을 가지고 있어 제안된 코프로세서는 재구성 정보와 프로그램 명령을 재사용할 수 있다. 따라서 주어진 연산흐름이 바뀌지 않는 한 명령어의 패치가 필요 없게 된다. 따라서 제안된 코프로세서는 프로

그램 메모리의 접속 시간을 크게 줄일 수 있게 되고, 부가적으로 메모리 접속에 따른 전력소모를 크게 줄일 수 있을 것이다.

#### IV. 구현 결과 및 성능 비교

앞서 제안한 기능별 연산블록에 대한 하드웨어를 기반으로 재구성가능 코프로세서 구조에 대한 검증을 수행하였다. 제안한 구조는 VHDL 코딩을 통해 시뮬레이션 하였으며, SEC 0.18 $\mu$ m 표준 셀 라이브러리를 사용하여 합성하였다. 곱셈/덧셈/뺄셈/비교 연산기의 경우 전체 게이트(Gate) 수는 약 17,000개, 비트 조작 연산기의 경우 전체 게이트(Gate) 수는 1,700개였으며, 최대 지연 경로는 3.84ns를 기록하였다. 제안한 연산블록은 재구성 가능한 프로세서 구조에서의 기능 및 성능검증을 위해 시뮬레이터를 통하여 구현 하였다.

그림 12는 LISATekTM을 이용한 시뮬레이터를 나타낸 것이다<sup>[13, 14]</sup>.

시뮬레이터를 이용하여 다양한 통신 알고리즘에 대한

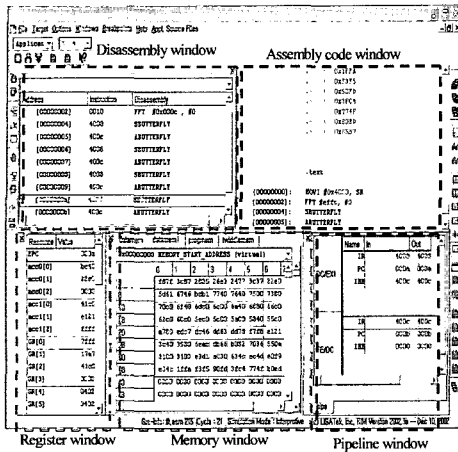


그림 12. 구현한 시뮬레이터  
Fig. 12. Simulator modeling.

표 1. 제안한 곱셈 및 덧셈 구조의 성능 평가  
Table 1. Performance comparisons for various operations.

Operation	SC140 (Operations/Cycle)	Proposed (Operations/Cycle)
Multiply	4	4
MAC	4	4
Complex multiply	1	1
Complex MAC	1	1.5
FFT butterfly	1	1.5
BM calculation (code rate : 1/2)	1	4
BM calculation (code rate : 1/3)	1	2
Trellis butterfly	2	2

\* 표의 숫자는 사이클당 연산수

기능 검증을 수행하였다. 먼저 곱셈/덧셈/뺄셈 구조의 성능을 측정 한 결과 표 1과 같이 나타났으며 이는 일반적인 4 MAC, 4 ALU 구조의 프로세서(StarCore SC140) 보다 사이클당 수행 수에서 동일하거나 향상된 성능을 얻을 수 있었다. 따라서 많은 연산량을 요구하는 터보 디코딩 연산 처리에 300MHz로 동작하는 스타코어의 경우 연산 능력의 1/3만 사용하여 384kbps 데이터 전송을 지원할 수 있기 때문에 제안한 구조의 경우 더 적은 하드웨어 자원을 사용하면서도 더 높은 연산처리로 터보 디코딩의 실시간 처리가 가능하다<sup>[15-17]</sup>.

예를 들어 SC140의 경우 2 사이클당 2개의 FFT 버터플라이의 연산이 가능한데 반해, 제안된 코프로세서의 경우 2 사이클당 3개의 FFT 버터플라이가 가능하다. 또

표 2. 비트 조작 연산기 사용에 따른 성능 비교  
Table 2. Performance comparisons of bit manipulations.

	SC140 [11]	TI 62x [12]	Proposed coprocessor
Operator	4 Shifters, 4 ALUs	4 Shifters, 4 ALUs	Bit Manipulation Unit
Convolution Encoding (Cycle) (IS-95, K=9, R=1/2, 192bits)	463	N.A.	152
Block Interleaving (Cycle) (16 * 6 bits)	414	N.A.	91
Scrambling (MIPS) (802.11a, 12Mbps)	N.A.	39 x 106	20 x 106
Convolution Encoding (MIPS) (802.11a, 12 Mbps)	N.A.	77 x 106	12 x 106

한 FFT 연산과 복소 MAC 연산의 경우 약 33%의 사이클 감소가 가능하고, 비터비 연산의 경우 37%의 사이클 감소가 가능하였다.

표 2는 제안한 비트조작 연산기와 명령어를 사용한 재구성 가능 코프로세서와 일반적인 DSP의 성능을 비교한 것이다. 일반적인 DSP의 경우 4개의 시프트 및 논리 연산기를 갖는 VLIW 구조임에도 제안된 비트조작 연산기를 사용한 것이 더 효율적으로 나타났다. 스타코어와 비교할 경우 IS-95 표준의 구속장 9, 부호율 1/2인 길쌈부호화 수행 시 67%, 16\*6 비트 크기의 블록 인터리빙 시 78%의 연산 사이클을 감소시킬 수 있다. 또한 TI 62x와 비교하면, IEEE 802.11a 표준에서 데이터 전송율이 12Mbit/s일 경우 스크램블링은 48%, 길쌈 부호화는 84%의 연산량 감소 효과를 얻을 수 있었다<sup>[18-20]</sup>.

### V. 결 론

본 논문은 서로 다른 연산 알고리즘을 지원하는 다양한 통신표준에 대하여 유연하게 사용 가능한 재구성 가능 코프로세서를 제안하였다. 제안한 코프로세서는 통신 시스템에 보편적으로 사용되는 스크램블링, 인터리빙, 길쌈 부호화, FFT 등 거의 모든 통신 연산에 사용할 수 있다. 제안된 코프로세서는 VHDL로 설계되었으며 SEC 0.18 $\mu$ m 표준셀 라이브러리로 합성하여, 게이트 카운트 약 35,000에 최대 지연 시간 3.84ns를 보였다. 제안된 구조의 설계 및 성능평가 결과 FFT 및 복소 MAC 연산에서 약 33%의 연산 사이클 감소, 비터비 연산에서 약 37%의 연산 감소, IEEE 802.11a 표준 (12Mbps 전송)의



스크램블링 (48%), 길쌈 부호화 (84%)의 연산량 감소 효과가 있었으며, 하드웨어 크기 또한 기존 범용 DSP의 연산구조에 비해 크지 않는 등 차세대 통신 시스템을 위한 재구성가능 코프로세서 구조로 적합할 것으로 예상된다.

## 참 고 문 헌

- [1] C. L. Wang and C. H. Chang, "A novel DHT-based FFT/IFFT processor for ADSL transceivers," in *Proc. IEEE Int. Symp. Circuits Syst.*, 1999, pp. 51-54.
- [2] VDSL Alliance, *VDSL Alliance Draft Standard Proposal*, April 1999.
- [3] IEEE Std 802.11a-1999, *Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications*, Sep. 1999.
- [4] J. R. Choi, S. B. Park, D. S. Han, and S. H. Park, "A 2048 complex point FFT architecture for digital audio broadcasting system," in *Proc. IEEE Int. Symp. Circuits Syst.*, 2000, vol. V, pp. 693-696.
- [5] DVB, *Framing Structure, Channel Coding and Modulation for Digital Terrestrial Television*, ETSI EN 300 744 v1.4.1, Jan. 2001.
- [6] Wiesler and F. K. Jondral, "A Software Radio for Second and Third Generation Mobile Systems," *IEEE Trans. on Veh. Tech.*, vol. 51, pp. 738-748, July 2002.
- [7] J. H. Lee, S. H. Jeong and M. H. Sunwoo, "Application-specific DSP architecture for OFDM modem systems," in *Proc. IEEE Workshop on Signal Processing Syst. (SIPS'2003)*, Aug. 2003.
- [8] F. Y. Kuo and C. W. Ku, "Software radio based re-configurable correlator/FIR filter for CDMA/TDMA receiver," in *Proc. IEEE Int. Symp. on Circuits and Systems*, Geneva, Switzerland, May 2000, vol. 1, pp. 112-115.
- [9] A. Chass, A. Gubeskys and G. Kuts [Motorola Semiconductor Israel Ltd.], "Efficient Software Implementation of the MAX-Log-MAP Turbo Decoder on the StarCore SC140 DSP".
- [10] SDR Forum, SDR Forum document number SDRF-02-I-0006-V0.10.
- [11] A. Wiesler and F. K. Jondral, "A software radio for second and third generation mobile systems," *IEEE Tran. On Veh. Tech.*, vol. 51, pp. 738-748, July 2002.
- [12] S. D. Kim, S. H. Jeong, M. H. Sunwoo, and K. H. Kim, "Novel Bit Manipulation Unit for Communication Digital Signal Processors," in *Proc. IEEE Int. Symp. on Circuits and Systems*, Vancouver, Canada, May 2004.
- [13] J. H. Lee, M. H. Sunwoo, and J. S. Lee, "Design of application-specific instructions and hardware accelerator for Reed-Solomon codec," *EURASIP JASP*, no. 13, pp. 1346-1354, Dec. 2003.
- [14] C. K. Chen, P. C. Tseng, Y. C. Chang, and L. G. Chen, "A digital signal processor with programmable correlator array architecture for third generation wireless communication system," *IEEE Trans. Circuit Syst. II*, vol. 48, pp. 1110-1120, Dec. 2001.
- [15] SC140 Functional Libraries, Motorola Inc. [Online]. Available: <http://www.motorola.com>.
- [16] J. Glossner, J. Moreno, M. Moudgill, J. Derby, E. Hokenek, D. Meltzer, U. Shvadron, and M. Ware, "Trends in compilable DSP architecture," in *Proc. IEEE Workshop on Signal Processing Syst.: Design and Implementation*, Oct. 2000, pp. 181-199.
- [17] K. Masselos, S. Blianas, and T. Rautio, "Reconfigurability requirements of wireless communication systems," in *Proc. IEEE Workshop on Heterogeneous Reconfigurable Systems on Chip*, April 2002.
- [18] E. Sereni, S. Culicchi, V. Vinti, E. Luchetti, S. Ottaviani, and M. Salvi, "A Software RADIO OFDM Transceiver for WLAN applications," Electronic and Information Engineering Department, University of Perugia, Italy, 2001.
- [19] *SC140 DSP Core Reference Manual*, Motorola Semiconductors Inc., Denver, CO, 2001.
- [20] *TMS320C62xx User's Manual*, Texas Instruments Inc., Dallas, TX, 2000.
- [21] Texas Instruments, Inc [Online]. Available: <http://www.ti.com>
- [22] Motorola, Inc [Online]. Available: <http://www.motorola.com>
- [23] Tensilica, Inc [Online]. Available: <http://www.tensilica.com>

---

 저 자 소 개
 

---



정 철 운(정회원)  
 2003년 아주대학교 전자공학  
 학사 졸업  
 2005년 아주대학교 전자공학  
 석사 졸업  
 2005년~현재 LG전자 DM  
 연구소 연구원

<주관심분야 : 통신알고리즘, VLSI 설계,  
 Digital Circuit Design>



선우 명 훈(평생회원)  
 1980년 서강대학교 전자공학  
 학사 졸업  
 1982년 한국과학기술원 전기전자  
 공학 석사 졸업  
 1982년 ~ 1985년 한국전자통신연  
 구소(ETRI)연구원

1985년~1990년 Univ. of Texas at Austin 전자  
 공학 박사 졸업

1990년~1992년 Motorola, DSP Chip Operation  
 (미국)

1992년 8월~현재 아주대학교 전자공학부 교수  
 <주관심분야 : VLSI 및 SoC Architecture, 통신  
 멀티미디어용 DSP 칩 및 SoC 설계>